Programme Structure

School of Engineering & Technology

B. Tech. CSE(Hons) /B. Tech. CSE(Hons) with Specialization AI&ML

Programme Code: ET0101

Batch: 2024-2028

SCHOOL OF ENGINEERING AND TECHNOLOGY

Program: B. Tech. CSE (Hons) AY: 2025-26

Batch: 2024-28 **Third Semester**

S. No	Status	Paper Code	Subjects	Study Scheme Lec / Week			Hours	Credits	CIE	ESE	Total	Pass Marks
				L	T	P						
1	CC-1	B010123301	Web Designing	3	1	0	4	3	50	50	100	40
2	CC-2	B010123302	Data Structures using C	3	1	0	4	3	50	50	100	40
3	CC-3	B010123303 / B010223303 / B010323303 / B010423303	Discrete Structure & Theory of Logic/ Mathematical Foundations for Artificial Intelligence / Mathematical Foundations for Artificial Intelligence / Information Theory for Cyber Security	3	1	0	4	3	50	50	100	40
4	DSC(minor)-1	B010123304	Object Oriented Programming using C++	4	0	0	4	4	50	50	100	40
5	MDC-1	B010123305	Computer Organization & Architecture	3	1	0	4	3	50	50	100	40
6	AEC-1	BSGUAE2406	Team Building & Leadership	2	0	0	2	2	50	50	100	40
7	SEC-1	BSGUSE2411	Advanced Excel	2	0	0	2	2	50	50	100	40
8	CC-4	B0101230351	Web Designing Lab	0	0	2	2	1	60	40	100	40
9	CC-5	B010123352	Data Structures Lab	0	0	2	2	1	60	40	100	40
10	CC-6	B010123354	OOPS with C++ Lab	0	0	2	2	1	60	40	100	40
11	SEC-2	BSGUSE2461	Advanced Excel Lab	0	0	2	2	1	60	40	100	40
				20	4	8	32	24	590	490	1100	440

DETAILED 2nd YEAR CURRICULUM CONTENTS

Undergraduate Degree in Engineering &Technology

PROGRAM: B.Tech. CSE(H)

With Specialization **AI&ML**

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester-3rd**
Course Name- Web Designing			
A.Y 2025-26	**Course Code- B010123301**	**Batch-2024-28**	**CIE Marks- 50 (MM)**
Total Teaching Hours 45 hrs	**Total Credits- 03**		**ESE Marks- 50 (MM)**
Type of Course - Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To introduce the basic principles and planning processes involved in designing responsive and standards-compliant websites.
2. To develop competency in creating and structuring web pages using HTML, focusing on key elements such as text formatting, lists, tables, multimedia, and forms.
3. To provide in-depth knowledge of CSS for enhancing the presentation layer of web pages, including the use of advanced layout techniques, positioning, and responsive design practices.
4. To familiarize students with client-side scripting using JavaScript, enabling interactivity through event handling, form validation, and dynamic content manipulation.
5. To introduce the fundamentals of web hosting and search engine optimization (SEO), including domain registration, hosting control panels, FTP usage, and techniques for on-page SEO.

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
1	**Introduction :** Basic principles involved in developing a web site, Planning process , Domains and Hosting, Responsive Web Designing , Types of Websites (Static and Dynamic Websites), Web Standards and W3C recommendations, **Introduction to HTML:** What is HTML , HTML Documents, Basic structure of an HTML document , Creating an HTML document , Mark up Tags , Heading-Paragraphs , Line Breaks	09
2	**Elements of HTML:** HTML Tags., Working with Text , Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia, Working with Forms and controls	09
3	**Concept of CSS:** Creating Style Sheet, CSS Properties , CSS Styling(Background, Text Format, Controlling Fonts) , Working with block elements and objects , Working with Lists and Tables , CSS Id and Class, Box Model(Introduction, Border properties, Padding Properties, Margin properties) CSS Advanced(Grouping, Dimension, Display, Positioning, Floating, Align, Pseudo class, Navigation Bar, Image Sprites, Attribute sector) , CSS Color , Creating page Layout and Site Designs.	09
4	Introduction to Client Side Scripting , Introduction to Java Script , Javascript Types , Variables in JS, Operators in JS , Conditions Statements , Java Script Loops, JS Popup Boxes , JS Events , JS Arrays, Working with Arrays, JS Objects ,JS Functions , Using Java Script in Real time , Validation of Forms, Related Examples.	09
5	**Web Hosting:** Web Hosting Basics , Types of Hosting Packages, Registering domains , Defining Name Servers , Using Control Panel, Creating Emails in Cpanel , Using	09

	FTP Client, Maintaining a Website **Concepts of SEO :** Basics of SEO, Importance of SEO, Onpage Optimization Basics	

Course Outcomes

CO 1 Understand principle of Web page design and about types of websites
CO 2 Visualize and Recognize the basic concept of HTML and application in web designing.
CO 3 Recognize and apply the elements of Creating Style Sheet (CSS).
CO 4 Understanding the basic concept of Java Script and its application.
CO 5 Introduce basics concept of Web Hosting and apply the concept of SEO.

Text books:

1. Steven M. Schafer, "HTML, XHTML, and CSS Bible, 5ed", Wiley India
2. Ian Pouncey, Richard York, "Beginning CSS: Cascading Style Sheets for Web Design", Wiley India

Reference books:

1- Web Design with HTML, CSS, JavaScript and jQuery Set by Jon Duckett, publisher Willey.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester-3rd**
Course Name- Computer Organization & Architecture			
A.Y 2025-26	**Course Code- B010123305**	**Batch-2024-28**	**CIE Marks- 50 (MM)**
Total Teaching Hours 45 hrs	**Total Credits- 03**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To introduce the basic functional units of a digital computer system and explain how registers, buses, and memory interact to support processor operations.
2. To understand the design and functioning of Arithmetic and Logic Units (ALU).
3. To explore the functioning of the control unit through various instruction formats, instruction cycle phases, pipelining, and control strategies including hardwired and microprogrammed control.
4. To analyze memory system architecture and hierarchy**,** covering RAM, ROM, cache, virtual memory, and auxiliary storage, with focus on design and performance optimization.
5. To examine the mechanisms and architecture of I/O systems, including interrupt handling, data transfer techniques, I/O interfaces, and communication protocols for peripheral devices.

UNIT-1	**Topics**	**No. of Teaching hours/ (Lecture)**
1	**Introduction**: Functional units of digital system and their interconnections, buses, bus architecture, types of buses and bus arbitration. Register, bus and memory transfer. Processor organization, general registers organization, stack organization and addressing modes.	**09**
2	**Arithmetic and logic unit:** Look ahead carries adders. Multiplication: Signed operand, multiplication, Booths algorithm and array multiplier. Division and logic operations. Floating point arithmetic operation, Arithmetic & logic unit design. IEEE Standard for Floating Point Numbers	**09**
3	**Control Unit:** Instruction types, formats, instruction cycles and sub cycles (fetch and execute etc), micro operations, execution of a complete instruction. Program Control, Reduced Instruction Set Computer, Pipelining. Hardwire and micro programmed control: micro programme sequencing, concept of horizontal and vertical microprogramming.	**09**
4	**Memory:** Basic concept and hierarchy, semiconductor RAM memories, 2D & 2 1/2D memory organization. ROM memories. Cache memories: concept and design issues & performance, address mapping and replacement Auxiliary memories: magnetic disk, magnetic tape and optical disks Virtual memory: concept implementation.	**09**
5	**Input / Output**: Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions. Modes of Data Transfer: Programmed I/O, interrupt initiated I/O and Direct Memory Access., I/O channels and processors. Serial Communication: Synchronous & asynchronous communication, standard communication interfaces.	**09**

Course Outcomes

CO 1 Study of the basic structure and operation of a digital computer system.

CO 2	Analysis of the design of arithmetic & logic unit and understanding of the fixed point and floating-point arithmetic operations.
CO 3	Implementation of control unit techniques and the concept of Pipelining.
CO 4	Understanding the hierarchical memory system, cache memories and virtual memory.
CO 5	Understanding the different ways of communicating with I/O devices and standard I/O interfaces.

Text books:

1. Computer System Architecture - M. Mano
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky Computer Organization, McGraw-Hill, Fifth Edition, Reprint 2012
3. John P. Hayes, Computer Architecture and Organization, Tata McGraw Hill, Third Edition, 1998.

Reference books:

1. William Stallings, Computer Organization and Architecture-Designing for Performance, Pearso Seventh edition, 2006.
2. Web Design with HTML, CSS, JavaScript and jQuery Set by Jon Duckett, publisher Willey.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester-3rd**
Course Name- Data Structures using C			
A.Y 2025-26	**Course Code- B010123302**	**Batch-2024-28**	**CIE Marks- 50 (MM)**
Total Teaching Hours 45 hrs	**Total Credits- 03**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To introduce fundamental concepts of algorithms and data structures.
2. To develop skills in implementing linear data structures such as stacks and queues.
3. To provide knowledge of searching and sorting algorithms.
4. To explore the representation and traversal of graphs.
5. To impart a comprehensive understanding of tree data structures.

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
1	**Introduction:** Basic Terminology, Elementary Data Organization, Built in Data Types in C. Algorithm, Efficiency of an Algorithm, Time and Space Complexity, Asymptotic notations: Big Oh, Big Theta and Big Omega, Time-Space trade-off. Abstract Data Types (ADT) Arrays: Definition, Single and Multidimensional Arrays, Representation of Arrays: Row Major Order, and Column Major Order, Derivation of Index Formulae for 1-D,2-D,3-D and n-D Array Application of arrays, Sparse Matrices and their representations. Linked lists: Array Implementation and Pointer Implementation of Singly Linked Lists, Doubly Linked List, Circularly Linked List, Operations on a Linked List. Insertion, Deletion, Traversal, Polynomial Representation and Addition Subtraction & Multiplications of Single variable & Two variables Polynomial.	**09**
2	**Stacks:** Abstract Data Type, Primitive Stack operations: Push & Pop, Array and Linked Implementation of Stack in C, Application of stack: Prefix and Postfix Expressions, Evaluation of postfix expression, Iteration and Recursion- Principles of recursion, Tail recursion, Removal of recursion Problem solving using iteration and recursion with examples such as binary search, Fibonacci numbers, and Hanoi towers. Queues: Operations on Queue: Create, Add, Delete, Full and Empty, Circular queues, Array and linked implementation of queues in C, Dequeue and Priority Queue.	**09**
3	**Searching:** Concept of Searching, Sequential search, Index Sequential Search, Binary Search. Concept of Hashing & Collision resolution Techniques used in Hashing. Sorting: Insertion Sort, Selection, Bubble Sort, Quick Sort, Merge Sort, Heap Sort and Radix Sort.	**09**
4	**Graphs:** Terminology used with Graph, Data Structure for Graph Representations: Adjacency Matrices, Adjacency List, Adjacency. Graph Traversal: Depth First Search and Breadth First Search, Connected Component, Spanning Trees, Minimum Cost Spanning Trees: Prims and 08 Kruskal algorithm. Transitive Closure and Shortest	**09**

	Path algorithm: Warshal Algorithm and Dijikstra Algorithm.	
5	**Trees:** Basic terminology used with Tree, Binary Trees, Binary Tree Representation: Array Representation and Pointer(Linked List) Representation, Binary Search Tree, Strictly Binary Tree ,Complete Binary Tree . A Extended Binary Trees, Tree Traversal algorithms: Inorder, Preorder and Postorder, Constructing Binary Tree from given Tree Traversal, Operation of Insertation , Deletion, Searching & Modification of data in Binary Search . Threaded Binary trees, Traversing Threaded Binary trees. Huffman coding using Binary Tree. Concept & Basic Operations for AVL Tree , B Tree & Binary Heaps.	**09**

Course Outcomes

CO 1 Describe how arrays, linked lists, stacks, queues, trees, and graphs are represented in memory, used by the algorithms and their common applications.

CO2 Discuss the computational efficiency of the sorting and searching algorithms.

CO3 Implementation of Trees and Graphs and perform various operations on these data structure.

CO4 Understanding the concept of recursion, application of recursion and its implementation and removal of recursion.

CO5 Identify the alternative implementations of data structures with respect to its performance to solve a real world problem.

Text Books

1. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein, “Data Structures Using C and C++”, PHI Learning Private Limited, Delhi India
2. Horowitz and Sahani, “Fundamentals of Data Structures”, Galgotia Publications Pvt Ltd Delhi India.
3. Lipschutz, “Data Structures” Schaum’s Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd.
4. Thareja, “Data Structure Using C” Oxford Higher Education.
5.AK Sharma, “Data Structure Using C”, Pearson Education India.
6. P. S. Deshpandey, “C and Data structure”, Wiley Dreamtech Publication.

Reference Books

7. R. Kruse etal, “Data Structures and Program Design in C”, Pearson Education.
8. Jean Paul Trembley and Paul G. Sorenson, “An Introduction to Data Structures with applications”, McGrawHill.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester-3rd**
Course Name- Object Oriented Programming with C++			
A.Y 2025-26	**Course Code- B010123304**	**Batch-2024-28**	**CIE Marks- 50 (MM)**
Total Teaching Hours 50 hrs	**Total Credits- 04**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To introduce the fundamental concepts of Object-Oriented Programming (OOP).
2. To provide a comprehensive understanding of the C++ programming language.
3. To enable students to design and implement programs using classes and objects in C++.
4. To explore the concepts of inheritance and polymorphism in C++.
5. To develop skills for handling file operations in C++.

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
1	**Introduction** Introduction to object oriented programming: Basic Concepts of OOP, Benefits of OPP, Application of OOP, Class, Object, Inheritance Super Class, Sub Class, Overriding, Overloading, Encapsulation, Polymorphism, Abstraction, Interfaces, and Abstract Class.	**10**
2	**Introduction to C++** What is C++, A simple C++ Program, More C++ statements, Structure of C++ Program. Tokens, Expression and controls Structures Tokens , Keywords, Identifiers and Constants, C++ data types, Variables: Declaration, Dynamic initialization of variables, Reference variables, Operators in C++ : Scope resolution operator, Member deferencing Operators, Memory Management Operators, Manipulators, Type cast operators, Expressions and Control Structures. Functions The main() function, Function Prototyping, Call by reference, Return by reference, Inline function, Function Overloading.	**10**
3	**Classes and Objects** Introduction, Specifying a Class, Defining member Functions, C++ Program with Class, Nesting of Member functions, Private member functions, Memory Allocation for Objects, Static Data members, Static Member Functions, Arrays within a Class, Arrays of Objects, Objects as Function Arguments, Friendly Functions, Returning Objects. Pointers: Declaration and initializing, Manipulation of pointers, pointers Expression and Pointer Arithmetic, Pointer with Arrays, Arrays of Pointers, Pointers to objects, this pointers, Arrays of Pointers to Objects. Constructors and Destructors Constructors, Parameterized Constructors, Multiple Constructors in a class, Copy constructor, Destructors. Operator overloading Defining Operator Overloading, Overloading Unary Operators, Overloading Binary Operators, Type Conversions.	**10**
4	**Inheritance and Polymorphisms** Introduction, Defining Derived Classes, Single inheritance, Multiple inheritance, Hierarchical inheritance, Multilevel inheritance, Hybrid inheritance, Virtual Base	**10**

	Classes, Polymorphism, static and dynamic binding, Constructor in Derived Classes, Pointers to Derived Classes, Virtual Functions, Pure Virtual Functions.	
5	**I/O Operations and Files** C++ Stream Classes, Unformatted I/O Operations, Formatted I/O operations, Classes for File Streams, Opening and Closing a File : open() and close() functions, Manipulators of File Pointers : seekg(), seekp(), tellg(), tellp() functions, Sequential Input and output Operations : put (), get(), write(), read() functions, Error handling File Operations : eof(), fail(), bad(), good( ).	**10**

Course Outcomes

CO 1 Develop the object-oriented programming concepts.
CO2 Implement Structure of C++ Programs.
CO3 Defining member Functions in C++ Program with Class and objetcs.
CO4 Defining Derived Classes and implement inheritance.
CO5 Implement file handling in C++.

Text Books

1. E. Balagurusamy - Object Oriented Programming with C++ - TMH.

Reference Books:

1. Robert Lafore - Object Oriented Programming in Microsoft C++ - Galgotia.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester-3rd**
Course Name- Mathematical Foundations for Artificial Intelligence			
A.Y 2025-26	**Course Code- B010223303**	**Batch-2024-28**	**CIE Marks- 50 (MM)**
Total Teaching Hours 45 hrs	**Total Credits- 03**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To understand the fundamentals of descriptive statistics including data visualization, central tendency, dispersion, and correlation.
2. To learn the principles of probability theory, probability distributions, and foundational theorems for statistical inference.
3. To apply inferential statistical methods such as hypothesis testing, confidence intervals, and regression analysis.
4. To explore concepts in vector spaces, inner product spaces, and orthonormal basis for mathematical modeling.
5. To analyze linear transformations, eigenvalues, and matrix diagonalization for solving advanced algebraic problems.

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
1	**Descriptive Statistics & Probability** Diagrammatic representation of data, measures of central tendency, measures of dispersion, measures of skewness and kurtosis, correlation, inference procedure for correlation coefficient, bivariate correlation, multiple correlations	09
2	**Inferential Statistics & Regression** Measures of probability, conditional probability, independent event, Bayes' theorem, random variable, discrete and continuous probability distributions, expectation and variance, markov inequality, chebyshev's inequality, central limit theorem	09
3	**Pseudo-Random Numbers & Monte Carlo Integration** Sampling & confidence interval, Inference & Significance. Estimation & Hypothesis Testing, Goodness of fit, Test of Independence t-test/z-test (one sample, independent, paired), ANOVA, chi-square. Regression, Linear Methods for Regression Analysis	09
4	**Vector Spaces** Vector Space, Subspace , Linear Combination, Linear Independence, Basis, Dimension, Finding a Basis of a Vector Space , Coordinates, Change of Basis Inner Product Spaces- Inner Product, Length, Orthogonal Vectors, Triangle Inequality, Cauchy Schwarz Inequality, Orthonormal (Orthogonal) Basis, Gram-Schmidt Process.	09
5	**Linear Transformations** Linear Transformations and Matrices for Linear Transformation, Kernel and Range of a Linear Transformations, Change of Basis Eigenvalues and	09

	Eigenvectors- Definition of Eigenvalue and Eigenvector, Diagonalization , Symmetric Matrices and Orthogonal Diagonalization.	

Course Outcomes

CO 1 Demonstrate a sound understanding of Descriptive Statistics & Probability and their role and importance in artificial intelligence and machine learning

CO2 Demonstrate a good understanding of Sampling & Confidence Interval, Inference & Significance, Estimation and Hypothesis Testing,

CO3 Understanding of random number generation and Monte Carlo Integration.

CO4 Understanding of concepts of vector spaces.

CO5 Demonstrate a good understanding of linear transformations and matrices.

Text Books

1. S.C. Gupta & V.K. Kapoor, "Fundamentals of Mathematical Statistics", Sultan Chand & Sons
2. Sheldon M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", Academic Press.
3. Dudewicz, E.J., Mishra, S.N., "Modern Mathematical Statistics", Willy.
4. Purohit S. G., Gore S. D., Deshmukh S. K., "Statistics using R, Narosa.

Reference Books:

5. Rizzo, M. L., "Statistical Computing with R", Boca Raton, FL: Chapman & Hall/CRC Press.
6. Normal Maltoff, The Art of R programming, William.
7. Dalgaard, Peter, "Introductory statistics with R", Springer Science & Business Media.
8. M. D. Ugarte, A. F. Militino, A. T. Arnholt, "Probability and Statistics with R", CRC Press.
9. Kundu, D. and Basu, A., "Statistical computing – existing methods and recent developments", Narosa.
10. Gentle, James E., Härdle, Wolfgang Karl, Mori, Yuich, "Handbook of Computational Statistics", Springer.
11. Givens and Hoeting, "Computational Statistics", Wiley Series in Prob. and Statistics.
12. Elementary Linear Algebra by Ron Larson, 8th edition, Cengage Learning, 2017

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester- 3rd**
Course Name: Team Building & Leadership			
A.Y 2025-26	**Course Code- BSGUAE2406**	**Batch-2024-28**	**CIE Marks- 50 (MM)**
Total Teaching Hours 30 hrs	**Total Credits- 02**		**ESE Marks- 50 (MM)**
Type of Course- Theory			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. To understand the concept, significance, and development stages of effective teams in organizational settings.
2. To explore foundational leadership theories, styles, and their application in team environments.
3. To develop skills for building, leading, and sustaining high-performing and cross-functional teams.
4. To analyze team dynamics and apply conflict resolution and motivation techniques for team cohesion.
5. To evaluate team performance using structured tools and provide constructive feedback for continuous improvement.

UNIT-1	Topics	No. of Teaching hours/ (Lecture)
1	**Introduction to Team Building** Definition and importance of teams in organizations, Stages of team development: Forming, Storming, Norming, Performing, Adjourning, Characteristics of effective teams, Role of communication in team building.	**06**
2	**Leadership Fundamentals** Definition and significance of leadership in team settings, Leadership vs. Management, Leadership theories: Trait theory, Behavioral theory, Contingency theory, Leadership styles: Autocratic, Democratic, Laissez-faire, Transformational, and Transactional leadership.	**06**
3	**Building and Leading Teams** Team roles and responsibilities, Creating and sustaining high-performing teams, Leadership in diverse and cross-functional teams, Strategies for effective team leadership.	**06**
4	**Managing Team Dynamics** Understanding team dynamics and group behavior, Conflict resolution in teams, Motivating team members, Enhancing team collaboration and trust.	**06**
5	**Assessing Team Performance** Tools and techniques for evaluating team performance, Providing feedback and coaching, Continuous improvement in team processes, Case studies on successful team leadership.	**06**

Course Outcomes

CO 1 Identify and describe the key components of effective team building.

CO2 Analyze various leadership styles and their application in team management.

CO3 Demonstrate the ability to work collaboratively within a team, contributing to group goals.

CO4 Apply leadership skills in managing team dynamics and resolving conflicts.

CO5 Evaluate team performance and provide constructive feedback for improvement.

Text Books

1. Adair, J. (2009). Effective Team Building. Pan Books.
2. Northouse, P. G. (2018). Leadership: Theory and Practice. Sage Publications.
3. Lencioni, P. (2002). The Five Dysfunctions of a Team: A Leadership Fable. Jossey-Bass.

Reference Books

4. Blanchard, K. (2007). Leading at a Higher Level. FT Press.
5. Levi, D. (2015). Group Dynamics for Teams. Sage Publications.

.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester- 3rd**
Course Name: Advanced Excel			
A.Y 2025-26	**Course Code- BSGUSE2411**	**Batch-2024-28**	**CIE Marks- 25 (MM)**
Total Teaching Hours 30 hrs	**Total Credits- 02**		**ESE Marks- 25 (MM)**
Type of Course- Theory			**Total Marks-50 (MM)**

Course Objectives/Course Description

1. To apply built-in and custom conditional formatting, dynamic highlighting, and advanced lookup functions for efficient data visualization and tracking.
2. To implement advanced Excel functions such as OFFSET, INDEX-MATCH, and statistical formulas to create interactive HR dashboards and reports.
3. To utilize cloud features for collaborative reporting including PowerPoint/PDF exports, embed codes, permissions, and mail subscriptions.
4. To develop financial models using Excel's financial functions (e.g., PMT, IRR, XIRR, CUMPRINC) and perform forecasting and statistical analysis.
5. To analyze business data using descriptive and predictive analytics with macros, Power Query, dashboards, and VBA for automation.

UNIT-1	**Topics**	**No. of Teaching hours/ (Lecture)**
1	Conditional Formatting - Inbulit Conditional Formatting, Custom Conditional Formatting, Dynamic Search & Highlight, Partial Match Vlookup/ Hlookup/ Xlookup – 18 Scenario	**06**
2	Offset Function–Basic, Offset with Sum – Horizontal, Offset with Sum – Vertical, Offset with Match, Offset with Average, Offset with Countif Cloud Features – Edit Report, Export to PowerPoint & PDF, Embed Code Generation Setting, Manage Permissions & Content Pack, Mail Subscription, Quick Insights	**06**
3	Index & Match Function - Index Function Basic, Index Function with Area Number, Match Function Basic, Index & Match Functions as replacement to Vlookup Function, Index & Match Functions with Drop Down, SUMIF, AVERAGEIF and COUNTIF Application of Exc	**06**
4	Financial Modelling in Excel: PMT, PPMT, IPMT,IRR, MIRR, XIRR, FV, FVSCEDULE, PV, CUMPRINC, Forecasting, Financial Statements Forecasting, Statistical tools-Standard Deviation , Correlation, Regression , histogram, testing-z-test,t-test, chi square	**06**
5	Business Analytics, Use of Spread Sheet to anlayze data-Descriptive analytics and Predictive analytics. Macro, Designing Dashboard Power Query Protection New Functions of Excel 201 VBA	**06**

Course Outcomes

CO 1 To understand the concept of look-up functions in Excel.

CO2 To gain the practical knowledge of worksheet.
CO3 To gain the practical knowledge of Excel Function & Formulas.
CO4 To gain the practical knowledge of Financial modelling in Excel.
CO5 To gain the practical knowledge of Business Analytics in Excel.

References:

1.William Fischer ,Excel: Quick Start Guide from Beginner to Expert .
2.Grey Harvey, Excel 2019 All – in- One For Dummies.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester- 3rd**
Course Name: Advanced Excel Lab			
A.Y 2025-26	**Course Code- BSGUSE2461**	**Batch-2024-28**	**CIE Marks- 30 (MM)**
Total Teaching Hours 15 hrs	**Total Credits- 01**		**ESE Marks- 20 (MM)**
Type of Course- Practical			**Total Marks-50 (MM)**

Course Objectives/Course Description

1. Apply inbuilt and custom conditional formatting and advanced lookup functions (VLOOKUP, HLOOKUP, XLOOKUP) to perform dynamic data highlighting and real-time search.
2. Use OFFSET and MATCH functions with SUM, AVERAGE, and COUNTIF to create dynamic ranges and analytical reports in Excel.
3. Utilize Excel cloud capabilities including report editing, exporting, embedding, permission management, and automated mail subscriptions.
4. Design interactive HR dashboards using functions like INDEX-MATCH, SUMIF, AVERAGEIF, COUNTIF, and dropdowns for real-time analytics.
5. Build financial models and perform business analytics through financial functions (PMT, IRR, XIRR), forecasting, descriptive/predictive statistics, Power Query, macros, and VBA automation.

	Topics	**No. of Teaching hours/ (Lecture)**
	Apply inbuilt and custom conditional formatting; create dynamic highlight rules and use partial match lookups with `VLOOKUP`, `HLOOKUP`, `XLOOKUP`. • Practice 18 real-time scenarios using advanced lookup functions and conditional formatting for dynamic search and highlight.	**15**
	• Implement `OFFSET` function for dynamic ranges with sum (horizontal & vertical), average, and conditional counting. • Integrate `OFFSET` with `MATCH` and `COUNTIF` to create dynamic reports and analytical views.	
	• Use Excel cloud features: edit reports online, export to PowerPoint/PDF, generate embed codes, and manage mail subscriptions. • Explore content pack creation, permission settings, and gain insights using "Quick Insights" in Excel cloud.	
	• Create HR dashboards using `INDEX`, `MATCH`, `SUMIF`, `AVERAGEIF`, `COUNTIF`, dropdowns, and interactive components. • Replace `VLOOKUP` with `INDEX-MATCH`, and build dynamic reports with drop-down interactivity and data filters.	
	• Apply financial functions like `PMT`, `IRR`, `XIRR`, `FV`, `PV`, `CUMPRINC`, and perform financial forecasting. • Analyze business data using descriptive and predictive analytics, macros, Power	

	Query, Excel 2019 functions, and basic VBA automation.	

Course Outcomes

CO 1 To understand the concept of look-up functions in Excel.
CO2 To gain the practical knowledge of worksheet.
CO3 To gain the practical knowledge of Excel Function & Formulas.
CO4 To gain the practical knowledge of Financial modelling in Excel.
CO5 To gain the practical knowledge of Business Analytics in Excel.

References:

1.William Fischer ,Excel: Quick Start Guide from Beginner to Expert .
2.Grey Harvey, Excel 2019 All – in- One For Dummies.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester- 3rd**
Course Name: Data Structures Lab			
A.Y 2025-26	**Course Code- B010123352**	**Batch-2024-28**	**CIE Marks- 60 (MM)**
Total Teaching Hours 15 hrs	**Total Credits- 01**		**ESE Marks- 40 (MM)**
Type of Course- Practical			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. Implement and analyze various sorting algorithms such as Bubble, Insertion, Selection, Shell, Radix, and Quick Sort for efficient data organization.
2. Develop stack-based applications using arrays and linked lists including postfix evaluation, parenthesis balancing, and infix-to-postfix conversion.
3. Construct and manipulate different types of queues including ordinary, circular, priority, and double-ended queues using array and linked list implementations.
4. Create and operate on various types of linked lists (singly, circular, doubly) to perform insertion, deletion, traversal, searching, counting, reversing, and polynomial operations.
5. Apply tree and graph data structures including binary search trees, heaps, expression trees, and graph algorithms like BFS and MST (using Kruskal's/Prim's methods) for solving real-world problems.

UNIT-1	**Topics**	**No. of Teaching hours/ (Lecture)**
	Implementing Sorting Techniques: Bubble Sort, Insertion Sort, Selection Sort, Shell , Sort, Radix Sort, Quick sort.	**15**
	Implementing Stacks: Array implementation, Linked List implementation, Evaluation of post expression and balancing of parenthesis , Conversion of infix notation to postfix notation. Implementing Queue: Linked List implementation of ordinary queue, Array implementation c circular queue, Linked List implementation of priority queue, Double ended queue.	
	Implementing Linked List: Singly Linked Lists, Circular Linked List, Doubly Linked Lists : I Display, Delete, Search, Count, Reverse(SLL), Polynomial , Addition , Comparative study of	
	Implementing Trees: Binary search tree : Create, Recursive traversal: preorder, post order, in Search Largest , Node, SSmallest Node, Count number of nodes, Heap: Min Heap, Max Heap reheap Up, reheap Down, Delete , Expression Tree, Heapsort.	
	Implementing Graphs: Represent a graph using the Adjacency Matrix, BFS, Find the minimum spanning tree (using any method Kruskal's Algorithm or Prim's Algorithm).	

Course Outcomes

CO1: Apply and implement various sorting algorithms including Bubble, Insertion, Selection, Shell, Radix, and Quick sort to analyze time complexity and improve data organization.

CO2: Demonstrate proficiency in searching and hashing techniques, including linear/binary search, hashing methods, and collision resolution strategies to optimize data retrieval.

CO3: Design and implement stack-based operations and algorithms such as postfix evaluation, infix to postfix conversion, and parenthesis balancing using array and linked list representations.

CO4: Construct various queue structures — circular, priority, and double-ended queues — using array and

linked list implementations to manage sequential data efficiently.
CO5: Develop and manipulate dynamic data structures including linked lists, trees (BST, Heaps), and graphs, and apply traversal, searching, and spanning tree algorithms for advanced data operations.

Text Books

1. Aaron M. Tenenbaum, Yedidyah Langsam and Moshe J. Augenstein, "Data Structures Using C and C++", PHI Learning Private Limited, Delhi India
2. Horowitz and Sahani, "Fundamentals of Data Structures", Galgotia Publications Pvt Ltd Delhi India.
3. Lipschutz, "Data Structures" Schaum's Outline Series, Tata McGraw-hill Education (India) Pvt. Ltd.
4. Thareja, "Data Structure Using C" Oxford Higher Education.
5. AK Sharma, "Data Structure Using C", Pearson Education India.
6. P. S. Deshpandey, "C and Data structure", Wiley Dreamtech Publication.

Reference Books

7. R. Kruse etal, "Data Structures and Program Design in C", Pearson Education.
8. Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with applications", McGrawHill.

School Name- School of Engineering & Technology			
Program- B. Tech.CSE(Hons) with Specialization AI&ML			**Semester- 3rd**
Course Name: OOPS LAB WIth C++			
A.Y 2025-26	**Course Code- B010123354**	**Batch-2024-28**	**CIE Marks- 60 (MM)**
Total Teaching Hours 15 hrs	**Total Credits- 01**		**ESE Marks- 40 (MM)**
Type of Course- Practical			**Total Marks-100 (MM)**

Course Objectives/Course Description

1. **Implement functions with default arguments, friend and inline functions**, and demonstrate different parameter passing methods: call by value, address, and reference.
2. **Develop basic C++ classes and objects**, incorporating constructors, destructors, copy constructors, and various data members including arrays, pointers, constants, and static members.
3. **Apply compile-time polymorphism** using function and operator overloading for efficient and modular code design.
4. **Demonstrate run-time polymorphism** through various inheritance types, virtual functions, and virtual base classes to support dynamic behavior in programs.
5. **Utilize templates for generic programming**, creating reusable class and function components in C++.

	Topics	No. of Teaching hours/ (Lecture)
	Implement functions with default arguments, friend and inline functions; demonstrate parameter passing using call by value, address, and reference.	**15**
	Basic Class and Object Concepts Create simple C++ classes demonstrating constructors, destructors, and copy constructors; explore data members including arrays, pointers, constants, and static members.	
	Compile-Time Polymorphism Apply operator and function overloading to demonstrate compile-time polymorphism in object-oriented programming.	
	Run-Time Polymorphism Implement inheritance (single, multiple, multilevel, etc.), virtual functions, and virtual base classes to achieve dynamic polymorphism.	
	Templates and Reusability Utilize class and function templates to create reusable and generic code components in C++.	

Course Outcomes

CO1: Apply function concepts in C++ including default arguments, friend functions, inline functions, and parameter passing methods for modular programming.

CO2: Design and implement user-defined classes with various types of data members (primitive, arrays, pointers, constants, static) and manage object lifecycle using constructors and destructors.

CO3: Demonstrate compile-time polymorphism using function overloading and operator overloading techniques.

CO4: Implement run-time polymorphism through different forms of inheritance, virtual functions, and virtual base classes.

CO5: Develop reusable and generic code using class and function templates to enhance software modularity and flexibility.

Text Books

2. E. Balagurusamy - Object Oriented Programming with C++ - TMH.

Reference Books:

2. Robert Lafore - Object Oriented Programming in Microsoft C++ - Galgotia.